Improving Automatic Music Tag Annotation Using Stacked Generalization Of Probabilistic SVM Outputs

S. R. Ness, A. Theocharis,
G. Tzanetakis
Dept. of Computer Science
University of Victoria
PO Box 3055, STN CSC
Victoria, BC, CANADA
sness@sness.net

L. G. Martins
Portuguese Catholic University
School of Arts / Research Center for Science
and Technology in the Arts
Rua Diogo Botelho, no 1327
Porto, Portugal
lmartins@porto.ucp.pt

ABSTRACT

Music listeners frequently use words to describe music. Personalized music recommendation systems such as Last.fm and Pandora rely on manual annotations (tags) as a mechanism for querying and navigating large music collections. A well-known issue in such recommendation systems is known as the cold-start problem: it is not possible to recommend new songs/tracks until those songs/tracks have been manually annotated. Automatic tag annotation based on content analysis is a potential solution to this problem and has recently been gaining attention. We describe how stacked generalization can be used to improve the performance of a state-of-the-art automatic tag annotation system for music based on audio content analysis and report results on two publicly available datasets.

Categories and Subject Descriptors

H.3.1 [**Information Storage and Retrieval**]: Content Analysis and Indexing Methods

General Terms

Algorithms, Theory, Experimentation

Keywords

sound analysis, music information retrieval, tags, folksonomies, music recommendation

1. INTRODUCTION

Music information retrieval (MIR) is a research area that has been rapidly gaining momentum due to the widespread digital distribution of music. A central goal of MIR is to create systems that can efficiently and effectively retrieve songs from large databases of music content. There are various approaches to specifying queries. We focus on the approach used by personalized music recommendation systems such as Last.fm and Pandora, which is, essentially, to represent each track as a collection of manually annotated words (tags). Social tags are a key part of "Web 2.0" technologies and have become an important aspect of recommendation systems. Any semantically meaningful word can be used for this purpose—tags for music can represent a variety of different concepts including genre, instrumentation, emotions, geographic origins, social conditions etc. Games with a purpose are an exciting new way of collecting tags for a variety of multimedia annotation tasks [20] by harnessing volunteer users who perform the annotation as part of casual gaming.

A well-known issue in tag-based recommendation systems is known as the cold-start problem [12]: it is not possible to recommend new songs/tracks until those songs/tracks have acquired enough manual annotations. Recently, games-with-a-purpose have been shown to be an effective way of acquiring reliable tags for large number of multimedia items. Automatic tag annotation based on content analysis can be used to complement manual tag annotation.

In this paper we focus on automatic tag annotation of music tracks in which the music retrieval system learns a relationship between acoustic features and words from a dataset of annotated audio tracks. The resulting trained model can retrieve audio tracks based on lists of tags and can annotate unlabelled audio tracks with tags. Such systems can be used to both annotate novel audio content as well as retrieve relevant audio tracks from a database of unannotated tracks given a text-based query [18]. Similar approaches have been explored in the context of automatic image annotation [15].

Automatic audio annotation can be formulated as a multi-label classification problem. We describe a state-of-the-art automatic tag annotation system for music that utilizes audio feature extraction, the output of which is used to train a Support Vector Machine (SVM) with probabilistic class outputs, where each class corresponds to a tag. Stacked generalizaton is used in order to train a second level SVM classifier that exploits possible correlations between tags. We show that this significantly improves annotation performance on two publicly available music datasets with verified human annotations. There is no consistent terminology for stacked generalization and several other terms for similar approaches have been used in the literature of automatic content-based multimedia annotation and classification. In the following section we attempt to collect these different variations using common terminology and describe the differences and similarities between them and to the proposed approach.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
MM'09, October 19–24, 2009, Beijing, China.
Copyright 2009 ACM 978-1-60558-608-3/09/10 ...$10.00.

2. RELATED WORK

There is a large body of work in automatic image annotation [15]. Early work in audio annotation for music used web-documents associated with an artist for the text annotations [21]. There are several different approaches to collecting tags for music, each with advantages and disadvantages [17]. For example the Magnatagatune dataset used in this paper has been collected using TagATune [7], a game with a purpose. There has been a recent increase in interest in automatic audio tag annotation for individual music tracks as evidenced by the corresponding task in the Music Information Retrieval Evaluation Exchange (MIREX) [3], an annual event where different MIR algorithms are evaluated on a variety of tasks. One of the best performing systems used a probabilistic model with one tag-level distribution over the audio feature space for each word in the vocabulary [18]. The parameters of a tag-level Gaussian Mixture Model (GMM) are estimated using audio content from a set of training tracks that are positively associated with the tag. This system had the best performance in MIREX 2008 and is used below as a baseline for comparison with our proposed approach. Like our system, the output for a particular track is a tag affinity vector that can be thresholded for tag annotation. Support Vector Machines have been used with song-level features for automatic tag classification trained at different granularities (track, album, artist) [9]. Unlike our approach, individual SVM are trained separately for each tag using positive and negative samples. Another possibility is to use boosting of classifiers for automatic generation of social tags for music recommendation [4]. A classifier specifically designed for multi-label classification was used to classify music into emotions [14]. Unlike in our approach, these systems have no second stage to model relations between tags is employed.

Audio tag annotation can viewed as a problem of multi-label classification [16]. Our approach is to use a distribution classifier (a linear SVM with probabilistic outputs [11]) that can output a distribution of affinities (or probabilities) for each tag. This affinity vector can either be used directly for indexing and retrieval, or thresholded to obtain a binary vector with predicted tag associations for the particular track. The resulting affinity vector is fed into a second stage SVM classifier in order to better capture the relations between tags. This approach is a specialized case of stacking generalization [22], a method for the combination of multiple classifiers. Similar ideas have appeared in the literature under other terms such as anchor-based classification [1] and semantic space retrieval [13], but not necessarily in a multi-label tag annotation context. The general idea is to map the content-based features to a more semantically meaningful space, frequently utilizing external information such as web resources. Stacked generalization has been used for discriminative methods for multi-label classification in text retrieval [6] but using a vector of binary predictions for each label to model dependencies between them. The most closely relevant work is applied in improving multi-label analysis of music titles again using a second stage classifier on the binary predictions of the first stage classifiers which the authors term the correction approach [10]. To the best of our knowledge this is the first time the probabilistic output of SVM classifiers is used for multiple label classification for automatic audio annotation and possibly more generally content-based multimedia annotation.

3. AUTOMATIC TAG ANNOTATION

Figure 3.1 shows the flow of information for our proposed audio annotation system. For each track in the audio collection a feature vector is calculated based on the audio content. As each track might be annotated by multiple tags the feature vector is fed into the multi-class **Audio SVM** several times with different tags. Once all tracks have been processed, the linear SVM is trained and a tag affinity output vector **(TAV)** is calculated. The TAV can be used directly for retrieval and storage or converted to a tag binary vector **(TBV)** by some thresholding method. When stacked generalization is used, the tag affinity vector (TAV) is used as a semantic feature vector for a second round of training over the tracks using an affinity SVM which produces a stacked tag affinity vector **(STAV)** and a stacked tag binary vector **(STBV)**. The resulting predicted affinity and binary vector can be used to evaluate the effectiveness of the retrieval system using metrics such as Area under Receiver Operating Characteristic Curve **(AROC)** for the TAV and information retrieval measures for the TBV.

3.1 Problem Formulation

We begin by considering a vocabulary V that consists of $|W|$ unique words and that each "word" refers to a semantic concept, for example "techno", "rock", "hardcore" or "ambient". The goal of annotation is the find a set $W = w_1, \ldots, w_A$ of A words that are semantically meaningful and describe a query audio track s_q. The process of retrieval consists of ordering a set of songs $S = s_1, \ldots, s_R$ when one is given a list of query words W_q. If we describe each song as an annotation vector $y = (y_1, \ldots, y_{|V|})$ where $y_i > 0$ if w_i has a semantic association with the audio track, and $y_i = 0$ if it does not. These y_i are proportional to the strength of the semantic association and are thus called semantic weights. We then map these semantic weights to the range $\{0, 1\}$ and interpret them as the class labels. We can then represent a song s as $X = x_1, \ldots, x_T$ of T real-valued feature vectors, with each vector x_t representing audio features that have been extracted from a short section of the song. The data set D that we use is a collection of pairs of tracks and annotations $D = (X_1, y_1), \ldots, (X_{|D|}, y_{|D|})$.

Automatic audio tag annotation can be viewed as a special case of multi-label classification. Traditional *single-label* classification is concerned with learning from a set of examples that are associated with a single label l from a set of disjoint labels L, $|L| > 1$. If $|L| = 2$, then the learning problem is called *binary* classification, while if $|L| > 2$ then it is called a multi-class classification problem. In *multi-label* classification the examples are associated with a set of labels $Y \subset L$. In addition, and in contrast to other multi-label classification problems, tags are relatively sparse and therefore there is an imbalance between positive and negative examples for each tag.

3.2 Audio Feature Extraction and Stacked Classification

Each audio track is represented as a single feature vector. Even though much more elaborate audio track representations have been proposed in the literature we like the simplicity of machine learning and similarity calculation using single feature vectors per audio clip. It has been shown that such song-level features perform quite well [8].

The features used are Spectral Centroid, Roll-Off, Flux

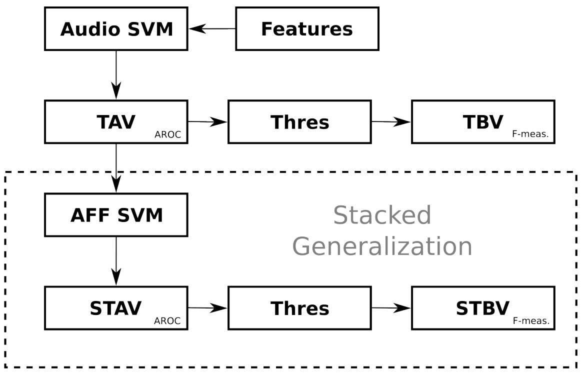

Figure 1: System flow diagram

and Mel-Frequency Cepstral Coefficients (MFCC). To capture the feature we compute a running mean and standard deviation over the past M frames:

$$m\Phi(t) = mean[\Phi(t-M+1),..,\Phi(t)] \quad (1)$$

$$s\Phi(t) = std[\Phi(t-M+1),..,\Phi(t)] \quad (2)$$

where $\Phi(t)$ is the original feature vector. Notice that the dynamics features are computed at the same rate as the original feature vector but depend on the past M frames (e.g. M=40, corresponding approximately to a so-called "texture window" of 1 second). This results in a feature vector of 32 dimensions at the same rate as the original 16-dimensional one. The sequence of feature vectors is collapsed into a single feature vector representing the entire audio clip by taking again the mean and standard deviation across the 30 seconds (the sequence of dynamics features) resulting in the final 64-dimensional feature vector per audio clip. A more detailed description of the features can be found in Tzanetakis and Cook [19].

For training the support vector machine classifier the feature vectors (one per audio track) are normalized so that the minimum of each feature is 0 and the maximum in 1 (Max/Min Normalization). The *Marsyas* audio processing framework (`http://marsyas.sness.net`) was used for the computation of the features. Both stages of classification (audio and stacked affinity) utilize a multi-class Support Vector Machine implemented as a collection of binary one-versus all discriminative classifiers. The libSVM software package is used for training and classification [2].

4. EXPERIMENTS

We tested the system on two publicly available audio datasets. The Computer Audition Lab 500 (CAL500) [18] dataset is a selection of 500 Western popular songs recorded by 500 different artists, from between 1958 and 2008. Each song is manually annotated with an appropriate subset of 135 total tags (including positive and negative tags), including 29 instruments, 22 vocal characteristics, 36 genres, 18 emotions, 15 preferred listening scenarios, and 15 concepts such as tempo and sound quality. Each song has at least 3 annotations, with a total of 1708 annotations in the collection. The Magnatagatune [7] dataset is a collection of 21642 songs and 188 tags. The songs were provided by Magnatune.com and FreeSound.org, and span the genres of classical, new age, electronica, rock, pop, world music, jazz, blues, heavy metal, and punk. Annotations for the files were collected via the TagATune game-with-a-purpose, in which two players were asked each to annotate a song. The players were then shown each other's annotations and asked to guess whether or not they had been listening to the same song.

	Accuracy Tag	F-measure Tag	AROC Tag	AROC Clip
BTL	0.842	0.258	0.68	0.78
Audio SVM	0.865	0.394	0.78	0.86
Affinity SVM	0.882	0.498	0.85	0.89

Table 1: CAL500 Evaluation Metrics

Table 2: Magnatagatune : Audio and Affinity SVM - Global evaluation metrics

	Precision	Recall	Accuracy	F-Score
Audio SVM	0.307	0.315	0.969	0.311
Affinity SVM	0.351	0.354	0.971	0.353

The results generated by the SVM algorithm are in the form of an affinity matrix, with one dimension representing all the songs in the collection, and one dimension representing the affinity of a particular tag for that song. This affinity matrix can be compared to a similarly constructed ground truth matrix via the Receiver Operating Characteristic (ROC) curve [5], which creates a curve by iteratively changing a cut-off level and plotting the resulting values. We can then integrate this curve to obtain the Area under Receiver Operating Characteristic curve (AROC). The values of AROC vary between 0 and 1, with larger values signifying better classifier performance. Precision, recall, accuracy and F-measure are also calculated in the standard way. We report these measures over both the entire (global) binary matrix as well as seperately for each tag and then averaged across tags. The per-tag average accuracy is a better measure as it is not biased by popular tags.

Table 1 shows various evaluation metrics comparing the performance of the best performing system in MIREX 2008 (LTB) [18]. To calculate these numbers, we obtained from the authors the predicted affinity matrix associating songs and tags for CAL500. The second line of the table shows the performance of the audio-based SVM system using song-level features. The third line shows the improvement in performance using the stacked generalization where the input to the second level classifier is the affinity vector predicted by the first level classifier. The same thresholding was applied in all cases. The threshold for each tag was chosen such that the number of testing songs associated with a given tag is proportional to the frequency in which that tag was applied to the training songs. All the results were obtained using 2-fold cross-validation and were not significantly different than using the training set for testing. This is expected given the challenging nature of multiple-label classification which makes over-fitting to the training data more unlikely.

Table 2 shows the results for both audio and stacked generalization using probabilistic SVM outputs for the Magnatagatune dataset and all tags. We believe this is the first time results are published for this dataset. The global evaluation metrics are biased towards popular tags, so can be misleading. As this dataset has many tags, we explore us-

# Tags	Precision	Recall	Accuracy	F-Score
20	0.417	0.688	0.856	0.516
30	0.345	0.669	0.862	0.452
40	0.370	0.381	0.910	0.375
50	0.328	0.337	0.919	0.332
100	0.189	0.195	0.947	0.192
all (188)	0.127	0.130	0.969	0.129

Table 3: Magnatagatune : Audio SVM - Per-tag evaluation metrics

# Tags	Precision	Recall	Accuracy	F-Score
20	0.418	0.691	0.856	0.518
30	0.346	0.671	0.862	0.453
40	0.394	0.397	0.914	0.395
50	0.369	0.372	0.923	0.371
100	0.259	0.262	0.951	0.260
all (188)	0.184	0.186	0.971	0.185

Table 4: Magnatagatune : Affinity SVM - Per-tag evaulation metrics

ing different number of tags for training the classifier. For example "30" means that the 30 most popular tags were used to train the classifier. Tables 3 and 4 show similar results by averaging the evaluation metrics across tags. This way tags that are not popular are as important as popular tags in terms of being predicted correctly. As can be seen, stacked generalization of probabilistic SVM outputs improves all per-tag evaluation metrics especially when all tags are considered. This is expected, as it can capture tag relations even among tags that might not be represented enough for accurate audio-based classification.

5. CONCLUSIONS

Stacked generalization of the probabilistic outputs of a Support Vector Machine classifier can be used to improve the performance of automatic audio tag annotation. The scheme is straightforward to implement and provides significant improvements over one stage classification using a variety of standard evaluation measures in two publicly available datasets. We believe that a similar approach could be used for other tasks such as automatic image annotation.

6. REFERENCES

[1] A. Berenzweig, D. P. W. Ellis, and S. Lawrence. Anchor space for classification and similarity measurement of music. In *Proc. of Int. Conf. on Multimedia and Expo (ICME)*, pages 29–32, 2003.

[2] C. Chang and C. Lin. *LIBSVM: a library for support vector machines*, 2001. Software available at http://www.csie.ntu.edu.tw/ cjlin/libsvm.

[3] S. J. Downie. The music information retrieval evaluation exchange (2005–2007): A window into music information retrieval research. *Acoustical Science and Technology*, 29(4):247–255, 2008.

[4] D. Eck, P. Lamere, T. Bertin-Mahieux, and S. Green. Automatic generation of social tags for music recommendation. In *Adv. in Neural Information Processing Systems*, volume 20, 2007.

[5] T. Fawcett. An introduction to ROC analysis. *Pattern Recognition Letters*, 27(8):861–874, June 2006.

[6] S. Godbole and S. Sarawagi. Discriminative methods for multi-labeled classification. In *Proc. Pacific-Asia Conf. on Knowledge Discovery and Data Mining*, 2004.

[7] E. L. M. Law, L. V. Ahn, R. B. Dannenberg, and M. Crawford. Tagatune: A game for music and sound annotation. In *Proc. Int. Conf. on Music Information Retrieval (ISMIR)*, 2007.

[8] M. Mandel and D. Ellis. Song-level features and support vector machines for music classification. In *Proc. Int. Conf. on Music Information Retrieval (ISMIR)*, 2005.

[9] M. Mandel and D. Ellis. Multiple-instance learning for music information retrieval. In *Proc. Int. Conf. on Music Information Retrieval (ISMIR)*, 2008.

[10] F. Pachet and P. Roy. Improving multilabel analysis of music titles: A large-scale validation of the correction approach. *Audio, Speech, and Language Processing, IEEE Transactions on*, 17(2):335–343, 2009.

[11] J. C. Platt. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In *Advances in Large Margin Classifiers*, pages 61–74. MIT Press, 1999.

[12] A. Schein, A. Popescul, L. Ungar, and D. Pennock. Methods and metrics for cold-start recommendations. In *Proc. ACM SIGIR Conf. on Research and Development in Information Retrieval*, 2002.

[13] M. Slaney. Mixtures of probability experts for audio retrieval and indexing. In *Multimedia and Expo, 2002. ICME '02. Proc. 2002 IEEE Int. Conf. on*, volume 1, pages 345–348 vol.1, 2002.

[14] K. Trohidis, G. Tsoumakas, G. Kalliris, and I. Vlahavas. Multilabel classification of music into emotions. In *Proc. Int. Conf. on Music Information Retrieval (ISMIR)*, 2008.

[15] C.-F. Tsai and C. Hung. Automatically annotating images with keywords: A review of image annotation systems. *Recent Patents on Computer Science*, 1:55–68, 2008.

[16] G. Tsoumakas and I. Katakis. Multi label classification: An overview. *Int. Journal of Data Warehouse and Mining*, 3(3):1–13, 2007.

[17] D. Turnbull, L. Barrington, and G. Lanckriet. Five approaches to collecting tags for music. In *Proc. Int. Conf. on Music Information Retrieval (ISMIR)*, 2008.

[18] D. Turnbull, L. Barrington, D. Torres, and G. Lanckriet. Semantic annotation and retrieval of music and sound effects. *Audio, Speech, and Language Processing, IEEE Transactions on*, 16(2):467–476.

[19] G. Tzanetakis and P. Cook. Musical Genre Classification of Audio Signals. *IEEE Trans. on Speech and Audio Processing*, 10(5), July 2002.

[20] L. von Ahn. Games with a purpose. *Computer*, 39(6):92–94, June 2006.

[21] B. Whitman and R. Rifkin. Musical query-by-description as a multiclass learning problem. In *In Proc. IEEE Multimedia Signal Processing Conf. (MMSP)*, pages 153–156, 2002.

[22] D. H. Wolpert. Stacked generalization. *Neural Networks*, 5:241–259, 1992.